



B.MATCH

# 认识 JavaScript



焦哥

# 目录

1 — JavaScript 基础

2 — JSON 数据

3 — 函数

4 — JS 存取数据

5 — TODO 例子

# 1

## JavaScript 基础

## JS 介绍

**JavaScript**, 通常缩写为**JS**, 是一种高级的, 解释执行的编程语言, 也可以称: 脚本语言、动态语言。

由ECMA(欧洲计算机制造商协会)通过ECMAScript实现语言标准化。它被世界上的绝大多数网站所使用, 也被世界主流浏览器(Chrome、IE、Firefox、Safari、Opera)支持。目前主流的版本是 ES6



## JS 原生数据类型

### ❖ **Number**: 数字类型

➤ 例子: 100、0.01

➤ 用法:

■ 加(+)、减(-)、乘(\*)、除(/)、整除(%)

### ❖ **String**: 字符串类型

➤ 例子: ""、'testing'、"3.14"、'name="myform"'、'This string \n has two lines'

➤ 用法:

■ 字符串拼接、长度、截取、大小写转化、查找

### ❖ **Boolean**: 布尔类型

➤ 例子: true、false

➤ 用法:

■ 判断真假

## JS 变量

- 
- ❖ **可变的:**
- 语法: `let key = "认识JS";`
  - 用法:
    - `key = "变化一下";`
    - `key = 123;`
    - `key = true;`
- ❖ **不可变的:**
- 语法: `const key = "不可变的值";`
- ❖ **打印变量:**
- 语法: `console.log(key);`

## JS 逻辑运算

### ❖ 计算数字:

#### ➤ 语法:

- `let key = 1+2+3;`
- `let key = 10/2;`
- `let key = (4+2)/2;`
- `key++;`
- `key--;`

### ❖ 条件语句:

#### ➤ 语法:

- 关系运算(>、>==、<、<==、===、!==)
- 逻辑运算(&&、||、!)

```
let a = 4;
let b = 2;
if ( a>b ){
    console.log('执行');
} else if ( a === b){
    console.log('执行相等');
} else {
    console.log('否则执行');
}
```

## JS 循环语句

### ❖ 数组:

#### ➤ 语法:

- `let array = [];`
- `let array = ["1",2,"数组"];`
- `array.push("添加一条数据");`

### ❖ for 语句:

#### ➤ 语法:

- `for(let i = 0 ; i < array.length ; i++ ){`  
    `// TODO`  
    `}`



```
let array = [0,1,2,3,4,5,6,7,8,9];  
  
for ( let i = 0 ; i < array.length ; i++){  
    console.log(array[i]);  
}
```

2

JSON 数据



**JSON(JavaScript Object Notation) 是一种轻量级的数据交换格式**

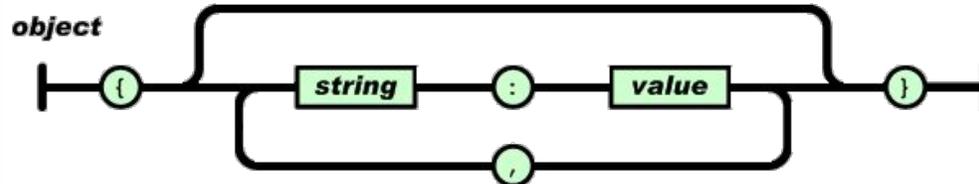
## JSON建构于两种结构

- “名称/值”对的集合
- 值的有序列表, 比如数组

## 对象

**对象是一个无序的“名称/值”对集合**

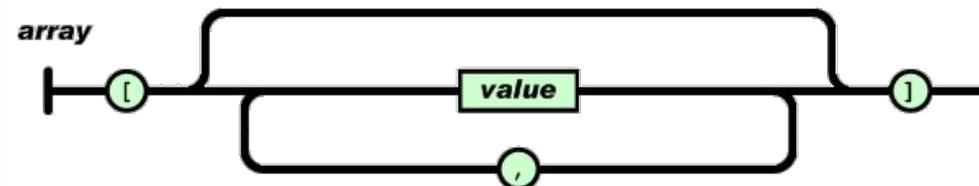
一个对象以“{”(左括号)开始, “}”(右括号)结束。每个“名称”后跟一个“:”(冒号);“名称/值”对之间使用“,”(逗号)分隔。



## 数组

**数组是值(value)的有序集合**

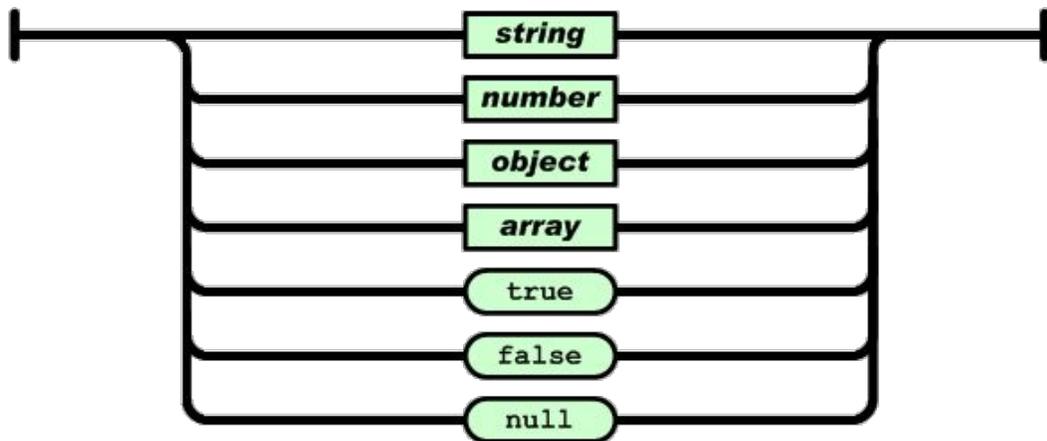
一个数组以“[”(左中括号)开始, “]”(右中括号)结束。值之间使用“,”(逗号)分隔。



## JSON的值类型

值(value)可以是双引号括起来的字符串(string)、数值(number)、true、false、null、对象(object)或者数组(array)。这些结构可以嵌套。

value



```
{
  "name": "小明",
  "age": 18,
  "isMale": true,
  "skills": [
    "Java",
    "SQL"
  ],
  "schools": [
    {
      "name": "杭州二中",
      "startTime": "2010-09"
    },
    {
      "name": "浙江大学",
      "startTime": "2013-09"
    }
  ]
}
```

## JS 操作 JSON

```
let json = {
  name: "小明",
  age: 18,
  isMale: true,
  skills: ["Java", "SQL"],
  schools: [
    {
      name: "杭州二中",
      startTime: "2010-09"
    },
    {
      name: "浙江大学",
      startTime: "2013-09"
    }
  ]
};
// json -> 字符串
const jsonString = JSON.stringify(json);
console.log(jsonString);

// 字符串-> json
const json2 = JSON.parse(jsonString);
console.log(json2.name+"："+json2.age);
```

3

函数

## 函数语法

函数是一堆代码的组合，也可以称为方法，它有自己的固定格式

- ❑ 方法体声明: function
- ❑ 方法名: 英文+数字组合，不能有空格
- ❑ 方法参数: 定义方法参数名，多个用逗号分隔
- ❑ 方法结构体声明: {}
- ❑ 方法返回值: return

```
/**
 * 定义一个实现加法的方法
 */
function add (param1,param2){
  // 执行加法
  const result = param1+param2;

  // 返回结果
  return result;
}

// 运行方法，得到结果
let result = add(1,2);
// 打印结果
console.log(result);
```

## 函数小例子

# 欢迎注册QQ

每一天，乐在沟通。

免费靓号

! 昵称不可以为空

可通过该手机号找回密码

同时开通QQ空间

我已阅读并同意相关服务条款和隐私政策 ▼

```
<input type="text" id="name">
<p id="tips"></p>
<button onclick="login()" >登陆</button>

function login() {
  let name = document.getElementById("name");
  let value = name.value;
  if(value){
    console.log("你输入的用户名是:"+value);
  }else{
    document.getElementById("tips").innerHTML = "用户名不能为空! ";
  }
}
```

<https://codepen.io/jiaoge/pen/zMWLJy?editors=1011>

## 函数小例子

### 奇偶数判断:

```
let oddEven = function(num){
  if(num%2 ==0){
    return true;
  }else{
    return false;
  }
}

if( oddEven(6) ){
  console.log('这是偶数');
}

if( !oddEven(5) ){
  console.log('这是奇数');
}
```

### 倒计时:

```
let num = 5;

let countdown = function() {
  console.log("倒计时: " + num);
  num--;
  if (num > 0) {
    setTimeout(countdown, 1000);
  }
};

countdown();
```

4

JS 存取数据

## localStorage

```
    // 创建数组
    let array = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9];
    // 获取存储对象
    const storage = window.localStorage;
    // 存储
    storage.setItem("array", JSON.stringify(array));
    // 获取值
    let a2 = storage.getItem("array");
    // 从字符串转化为 JSON
    a2 = JSON.parse(a2);

    // 移除数组
    storage.removeItem("array");
    // 清除所有数据
    storage.clear();
```

5

TODO 例子

TODO 数据例子

<https://codepen.io/jiaoge/pen/jQzemE>

<https://codepen.io/jiaoge/pen/MzVzBX?editors=1011>

# 总结

- 1 这节课学到了什么？
- 2 有哪些不明白的

# 作业

## **要求:**

实现按TODO过期时间排序的功能

## **作业提交地址:**

<https://quora.bmatch.tech/topic/5bf7c3314617e60035947425>